ST-1 C

No more "out of memory" errors! (was: how comment out audio...)

New Message | Reply | Date view | Thread view | Subject view | Author view | Attachment view

From: Todd Tannenbaum (*tannenba_at_engr.wisc.edu*)
Date: Tue Dec 06 1994 - 13:29:24 EST

- **Next message:** JABURNS_at_jcpenney.com: "Re: how comment out audio in system.ini"
- **Previous message:** Tim Vetter: "Re: how comment out audio in system.ini"
- **Next in thread:** Don Whiteside: "RE: No more "out of memory" errors! (was: how comment out audio...)"
- **Maybe reply:** Don Whiteside: "RE: No more "out of memory" errors! (was: how comment out audio...)"

>For those Windows 3.1x users who weren't following the earlier thread,
>commenting out the ThinkPad audio drivers from system.ini will save 182K of
>sub-1M memory. I used to get constant "out of memory" errors until I did
>this, now I haven't seen one for months.
>

I used to get lots of "out of memory" errors as well with Win 3.1 on my Thinkpad.

But not any more! AND, I still have the Thinkpad audio drivers loaded (as well as a ton of other junk) !!!!

How did I do it? I am using a small shareware utility called "MoreMem". It works great. I got it via anonymous ftp to ftp.cica.indiana.edu in the win3/utils directory (moremem.zip, or something almost identical to that). I liked it so much I even registered it (the author is asking $10; more than worth it to me!!).

Basically, the reason folks get "out of memory" errors is because they have run out of memory below 1meg. Every time you fire up a program in windows, windows needs to alloc a PSP in <1meg memory. A PSP takes about 400 bytes. If you are out of mem below 1meg, windows claims "out of memory" even if you have tons of >1meg RAM and resources left. besides drivers, any windows program which allocates "static", or non-relocateable memory, Windows gives it memory starting from the lowest address and grows upward. This means that lots of your precious below 1meg memory space is being used by windows programs/drivers which really do not care about getting memory below 1meg, they just don't want it swapped out....

When you put MoreMem in your windows startup folder, it immediately grabs memory from below 1meg and "reserves" it specifically for Windows PSPs.

(PSP = program segment prefix). since it is in your startup folder,
it can usually grab up a good sized chunk of memory below 1meg cuz other
windows apps have not yet chewed it up with non-relocating alloc calls to
windows.

I am running my Thinkpad 750C with Windows 3.1 with all ibm audio drivers,
ibm ezplay pcmcia drivers, ethernet, supertcp for windows with NFS/etc, Novell
DOS drivers, and a bunch of other crap. I can set MoreMEM to reserve
about 32 PSPs (which means I can run about 30 Windows programs
simultaneously before seeing the stupid "out of memory error"). It works.
I now routinely run
15-20 windows programs at once (i have 12 meg RAM). I could never do this
before Moremem (usually would get about 5 programs max before MoreMem).

I am in no way associated with MoreMem other than a happy user. :^).

p.s. I called up the author of MoreMem, and we brainstormed converting Moremem
>from a vanilla windows program into a windows .386 driver. This means Moremem
could do its magic even before other .386 drivers (like the ibm audio crap)
loaded. On my system, this would allow me to run over 100 Windows programs
at once (assuming i did not run out of RAM or windows gdi resources before
that!). However, with the current program I can run over 30 programs at once,
and that is certainly good enough for now...

Like I said, i got my copy via anonymous ftp to ftp.cica.indiana.edu. The
shareware version nags you by forcing you to type in a passcode every time
you start up windows. The registered version does not nag. Here
is the company info from the "About" window:

MoreMem
"A Utility to Prevent Out of Memory Errors When Loading Programs in Windows"
Moremem -Version 2.0a
by Daniel N. Woo
Copyright 1994
Gamma Research, Inc.
904 Bob Wallace Ave. Suite 212
Huntsville, AL 35801
Phone: (205) 533-7103
Compuserve: [73324,3027]
Cost is $10

Enjoy!

Todd Tannenbaum
CAE Model Advanced Facility
email: tannenba_at_engr.wisc.edu ph: 608-262-3118

- **Next message:** JABURNS_at_jcpenney.com: "Re: how comment out audio in system.ini"
- **Previous message:** Tim Vetter: "Re: how comment out audio in system.ini"
- **Next in thread:** Don Whiteside: "RE: No more "out of memory" errors! (was: how comment out audio...)"
- **Maybe reply:** Don Whiteside: "RE: No more "out of memory" errors! (was: how comment out audio...)"

New Message	**Reply**	**Date view**	**Thread view**	**Subject view**	**Author view**	**Attachment view**

This archive was generated by hypermail 2.1.3 : Thu Jan 23 2003 - 09:52:29 EST

Re: "Out of Memory" Error

John M. Goodman (*john@@agoodman.com*)
Thu, 04 Jun 1998 11:11:37 -0700

- **Messages sorted by:** [date][thread][subject][author]
- **Next message:** Wes Archer: "Goodbye"
- **Previous message:** John M. Goodman: "Re Helix' Hurricane and Dragon"
- **In reply to:** SeeLuke137@@aol.com: "SetMousePosition macro"

Curious--and frustrating, I am sure.

Yes, sometimes the first 640KB does matter. That certainly is true under Windows 3.x, and it could be so with Windows 95, though I was told it was less likely to be the case.

There are freeware and shareware programs that help keep "junk" out of that region and one of these might be compatible with your system and provide some relief. At least it might be worth a try.

Here is a portion of an article I wrote for Byte magazine (published in their Dec 95 issue) that will help you understand the situation:

>*The difficulties with Windows 3.x, running in 386-enhanced mode, arise*
>*from some deficiencies in how it manages memory. These deficiencies are*
>*part and parcel of the Windows architecture.*
>
> *1. Global DOS Memory*
>
>*Windows runs on top of DOS, which runs in real mode on an Intel x86*
>*processor (or equivalent CPU). In real mode the CPU can only address one*
>*megabyte of RAM. That makes the first megabyte of its memory address space*
>*very special. Windows itself and every Windows application must be able to*
>*get a small region of that first megabyte, or they simply cannot run. The*
>*Windows name for this memory is Global DOS Memory.*
>
>*Unfortunately, Windows does nothing to reserve this special memory region*
>*for the uses for which nothing else will do. When you start Windows, by*
>*default all the DLLs, for example, get loaded as low in memory space as*
>*possible. All of them would work just fine in extended memory, but until*
>*lower memory is too full, they won't go into higher addresses. If you have*
>*seen the message that starts "Insufficient memory to run this*
>*application..." the most likely cause was running out of Global DOS Memory.*
>*This is in many cases the most severe of Windows memory management*
>*problems.*

As I said, Win95 is *supposed* to have solved this, but its solutions are often less than one might wish them to be.

Here from that same Byte article is what I wrote about the
freeware/shareware programs I mentioned above:

>*_MoreMem_ by Daniel N. Woo, published by Gamma Research (shareware*
>*distributed as MMEM3.ZIP), _Windows Low Memory Saver_ published by Analysis*
>*& Management Enterprises (shareware distributed as DOSMEM.ZIP), and _1MB*
>*Fort_ by John McSorley which was published in PC Magazine in 1995 and is*
>*now freeware distributed as 1MBF.ZIP.*

...

>*The shareware and freeware programs (MoreMem, Windows Low Memory Saver, and*
>*1MB Fort) all solve the Global DOS Memory problem in similar, yet different*
>*ways. Each is effective for that purpose, but fails to handle any of the*
>*other problems.*

(The other problems alluded to in the preceding sentence are possible shortages of resources or total Windows memory.)

You can find these programs at the usual sites, such as:

http://www.shareware.com/

http://www.hotfiles.com/

http://www.davecentral.com/indexvc.html

http://www.sharewarejunkies.com/

http://www.tucows.com/us.html
(there is an international tucows site; I just don't have its URL handy)

http://www.download.com/

all of which I have listed in no particular order.

Let us know what the solution turns out to be, please. And good luck finding it.

John

At 01:22 AM 6/4/98 -0600, you wrote:
>
>*Hello again all,*
>
>*I tried something different tonight. I took control of the virtual memory*
>*away from Win95 and set it to be up to half a gig, but I still got the*
>*same "Out of Memory" error. Finally, I ran the Resource Meter, while I was*
>*attempting to start the program and it didn't show any changes at all.*
>*Resources don't seem to be the problem, so what else could it be? A friend*
>*of mine said that it was something to do with the amount of the first 640*
>*K that was available but to the best of my knowledge this doesn't matter.*

>*Yesterday, I decided to reinstall and retrain DNS. Everything went well*
>*until it was processing what I had read in the previous half an hour and*
>*then it stopped. I appreciate all of the suggestions, thank you. I really*
>*hope that someone can help me out of this problem.*
>
>*Have a great day, everyone!*
>
>*David Marshall*
>*dmarshal@@ucalgary.ca*

========

John M. Goodman, Ph.D., author of "Peter Norton's Inside the PC,"
Seventh Edition (Sams 1997, ISBN 0-672-31041-4)

========

- **Next message:** Wes Archer: "Goodbye"
- **Previous message:** John M. Goodman: "Re Helix' Hurricane and Dragon"
- **In reply to:** SeeLuke137@@aol.com: "SetMousePosition macro"

WEST

□ Generate Collection Print

L7: Entry 5 of 27 File: USPT Jun 18, 2002

US-PAT-NO: 6408313
DOCUMENT-IDENTIFIER: US 6408313 B1

TITLE: Dynamic memory allocation based on free memory size

DATE-ISSUED: June 18, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Campbell; David G.	Redmond	WA		
Christensen; Eric R.	Redmond	WA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Microsoft Corporation	Redmond	WA			02

APPL-NO: 09/ 212173 [PALM]
DATE FILED: December 16, 1998

INT-CL: [07] G06 F 17/30, G06 F 12/02

US-CL-ISSUED: 707/205; 711/118, 711/170
US-CL-CURRENT: 707/205; 711/118, 711/170

FIELD-OF-SEARCH: 707/3, 707/205, 707/2.3, 707/200, 707/206, 711/118-132, 711/133, 711/159, 711/170-173

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
□	4805097	February 1989	De Sanna	711/206
□	4812981	March 1989	Chan et al.	711/202
□	4961139	October 1990	Hong et al.	707/1
□	5517643	May 1996	Davy	709/105
□	5566315	October 1996	Milillo et al.	711/113
□	5668987	September 1997	Schneider	707/3
□	5893920	April 1999	Shaheen et al.	711/133
□	6128713	October 2000	Eisler et al.	711/159
□	6256645	July 2001	Mundy	707/205
□	6286016	September 2001	Heller et al.	707/206

ART-UNIT: 2171

PRIMARY-EXAMINER: Von Buhr; Maria N.

ABSTRACT:

A dynamic cache management mechanism determines the amount of memory available from a system on which a memory intensive application is running and dynamically manages the amount of cache for which real memory is associated or committed by the system. A background thread periodically queries the operating system to determine how much memory is available for use by applications. If the amount of memory, as identified in a free list is above a predetermined threshold, the application requests more memory for its use. If below the predetermined threshold, the application gives memory back to the operating system. The threshold used by the application to expand and shrink the memory it uses based upon system demand for memory.

21 Claims, 4 Drawing figures

WEST

□ Generate Collection | Print

L7: Entry 15 of 27 File: USPT Oct 3, 2000

US-PAT-NO: 6128713
DOCUMENT-IDENTIFIER: US 6128713 A

TITLE: Application programming interface enabling application programs to control allocation of physical memory in a virtual memory system

~~DATE ISSUED: Octo~~ber 3, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Eisler; Craig G.	Kirkland	WA		
Engstrom; G. Eric	Kirkland	WA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Microsoft Corporation	Redmond	WA			02

APPL-NO: 08/ 937059 [PALM]
DATE FILED: September 24, 1997

Not Prior Art

INT-CL: [07] G06 F 12/02

US-CL-ISSUED: 711/159; 711/160, 711/134, 711/136, 711/173, 709/104, 709/107, 709/108
US-CL-CURRENT: 711/159; 709/104, 709/107, 709/108, 711/134, 711/136, 711/160, 711/173

FIELD-OF-SEARCH: 711/203, 711/206, 711/208, 711/209, 711/159, 711/160, 711/133, 711/134, 711/136, 711/173, 709/104, 709/107, 709/108

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected | Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
□	4688167	August 1987	Agarwal	345/343
□	4967353	October 1990	Brenner et al.	711/160
□	5125086	June 1992	Perazzoli, Jr.	711/159
□	5386536	January 1995	Courts et al.	711/136
□	5394537	February 1995	Courts et al.	711/202
□	5499354	March 1996	Aschoff etal.	711/129
□	5572694	November 1996	Uchino	711/6
□	5606685	February 1997	Frandeen	711/117
□	5611064	March 1997	Maund et al.	711/209
□	5727178	March 1998	Pletcher et al.	711/202

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
0 620 523 A3	October 1994	EP	
0 620 523 A2	October 1994	EP	
0 713 176 A3	May 1996	EP	
0 713 176 A2	May 1996	EP	

OTHER PUBLICATIONS

PCT/US98/16800--International Search Report. Sep. 12, 1998.
"Method of Extending OS/2's Memory management to Recognize User Focus", IBM Technical Disclosure Bulletin, vol. 35, No. 1A, Jun. 1992, pp. 470-472.
V. Sohal, Reliable Memory Management for Real-Time Sytems, Electronic Design, vol. 44, No. 13, Jun. 1996, pp. 118, 120, 122, 124 XP000625394, see p. 120, right-hand col., line 7-page 124, left-hand col., line 39.
"Packing Variable-Sized Segments in the Swap File of a Paging-Based Virtual Memory System", IBM Technical Disclosure Bulletin, vol. 39, No. 3, Mar. 1996, pp. 301/302 XP000581702.
Adrian King, Intel Processor Architecture, Inside Windows 95, Microsoft Press, 1994, pp. 45-56.
Adrian King, A Tour of Chicago, Inside Windows 95, Microsoft Press, 1994, pp. 85-90.
Adrian King, The Base System, Inside Windows 95, Microsoft Press, 1994, pp. 121-131.
Device Driver Kit (DDK) for the Windows Operating System, Microsoft Corporation, 1995, Chaps. 1, 6, 11, 15.

ART-UNIT: 279

PRIMARY-EXAMINER: Nguyen; Hiep T.

ABSTRACT:

An application programming interface (API) enables application programs in a multitasking operating environment to control the allocation of physical memory in a virtual memory system. One API function enables applications to designate a soft page lock for code and data. The operating system ensures that the designated code and data is in physical memory when the application has the focus. When the application loses the focus, the pages associated with the code or data are released. When the application regains the focus, the operating system re-loads the pages into physical memory before the application begins to execute. The operating system is allowed to override the soft page lock where necessary. Another API enables applications to designate code or data that should have high priority access to physical memory,

without using a lock. This API enables the application to specifically control the likelihood that a piece of code or data will remain in physical memory by assigning a priority to the code or data that defines its priority relative to the priority of other code or data contending for the same physical memory.

25 Claims, 5 Drawing figures

CCLS: 711/159, 709/104, 709/107, 709/108, 711/134, 711/136, 711/160, 711/173